A Sierra Monitor Company

Driver Manual
(Supplement to the FieldServer Instruction Manual)

FS-8700-80 McQuay Micro Tech Open Protocol

APPLICABILITY & EFFECTIVITY

Effective for all systems manufactured after May 1, 2001

Driver Version:	**1.03**
Document Revision:	**2**

TABLE OF CONTENTS

1. McQuay Micro Tech Open Protocol Description

This document describes the FieldServer driver used to transfer data between a FieldServer and MicroTech® controllers. Transfers are done using the controller's Data Terminal Communications Protocol. In this document the protocol is referred to as the McQuay Micro Tech Open Protocol.

The McQuay Micro Tech Open Protocol driver allows the FieldServer to transfer data to and from devices over either RS-232 or RS-485 using McQuay Micro Tech Open Protocol. The FieldServer can emulate either a Server or Client.

When emulating a Client, the driver enables the FieldServer to request 'Everything' from the McQuay device as well as a number of advanced options which allow specific fields to be read or written and device scaling to be applied.

When emulating a Server, the driver provides an emulation of the byte memory of a device and responds to read and write requests.

The following **important points and limitations** should be noted before proceeding with this manual.

1. Arising from a feature of the McQuay Micro Tech Open Protocol is the peculiarity, that when a multi-byte value is written to a McQuay device, the write is done one byte at a time (one byte of data can be transferred per poll/response message pair) allowing for the possibility (at least for a short period) that the multi byte value is only partially correct until all the messages have been completed.
2. The vendor equipment is limited to a maximum of 9600 baud. Given that each message packet can only transfer one byte of data and that some data of interest is multi-byte, users of this protocol should expect low data transfer rates.
3. Port expansion is not supported for this driver.
4. A document identified as Ed15050 MicroTech Network Operations contains important information regarding the connection requirements of 3rd party equipment such as the FieldServer to a Microtech network. The document also defines a number of limitations for such connections. The following quotations from this document outline some of this information.

PC Connection
The PC connection to a MicroTech controller is through an available port A that is configured as TTY. It is best to connect the PC to a level-1 controller because data transmission is the fastest. However, a PC can be connected to any level-2 controller that does not have level-3 controllers connected to it or to any MicroTech level-3 controller. Regardless of where you connect the PC, you have access to the entire network.

You can connect two or more PCs to the network, but only one PC can be connected to a particular controller. The PC that you use most often should be connected to the level-1 controller for best performance. For example, you may have one PC that you use on site and another PC that you use off site. In this situation, you may want to connect the on-site PC to the level-1 controller and the modem for the off-site PC to a level-2 or level-3 controller.
If a PC is connected to a level-2 controller, a level-1 controller must poll that level-2 controller connected to the PC so that the PC has access to the entire network. Defining a level-2 controller in the level-1 controller Server list causes the level-1 controller to poll the level-2 controller.
***Port Configuration:** The communications port that the PC is connected to must be configured as a TTY port. The default port configuration for most MicroTech controllers sets port A as TTY. The port configuration is a software setting.*
A separate software setting defines the communications rate of each port. In most controllers, the default rate is 9600 bps.

***TTY:** A TTY port is used to connect a PC for monitoring purposes. It uses the RS-232C interface standard and the Data Terminal Communications protocol.*

5. The following quote from "MicroTech® Data Terminal Communication Packets, Open Protocol™ Data Communications Information Packet " highlights an important limitation of the protocol.

Data terminal communications may be sent via RS-232C or RS-485. Regardless of the electrical standard used for communications, this protocol is a single ended type (i.e., communications to one MicroTech controller maximum).

6. Each MicroTech controller is shipped from the factory with a unique job site password. The passwords are provided by the McQuay International representative at the time of startup.

7. The driver does not validate passwords when configured as a Server. This means that requests to read or write data will succeed even if the password supplied with the request is incorrect.

2. Driver Scope of Supply

2.1. Supplied by FieldServer Technologies for this driver

FieldServer Technologies PART #	DESCRIPTION
FS-8917-16	RJ45 to terminal connector cable.

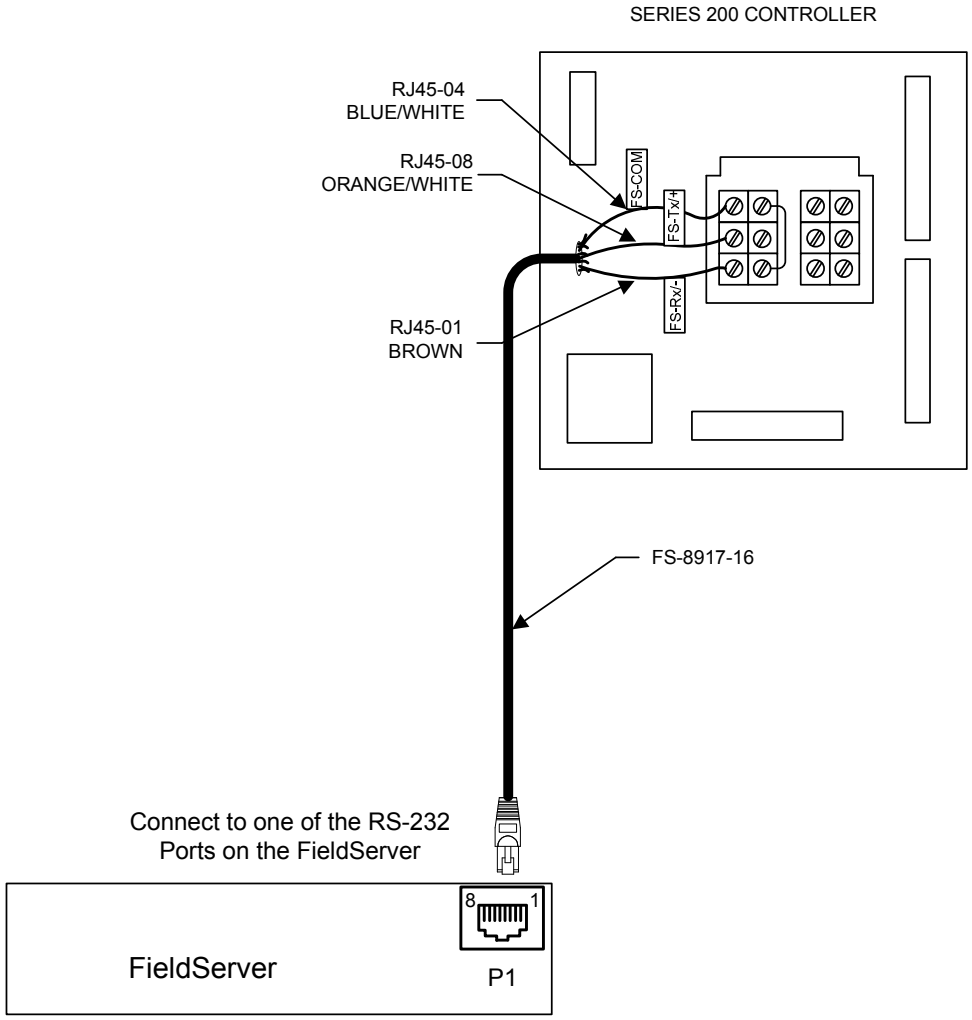
2.2. Provided by Supplier of 3rd Party Equipment

PART #	DESCRIPTION
	McQuay Motor Mount Connector for Port A connection

3. Hardware Connections

The FieldServer is connected to the McQuay device as shown below.

Configure the McQuay device according to manufacturer's instructions.



4. Configuring the FieldServer as a McQuay Micro Tech Open Protocol Client

For a detailed discussion on FieldServer configuration, please refer to the FieldServer Configuration Manual. The information that follows describes how to expand upon the factory defaults provided in the configuration files included with the FieldServer. (See ".csv" sample files provided with the FieldServer).

This section documents and describes the parameters necessary for configuring the FieldServer to communicate with a McQuay Micro Tech Open Protocol Server.

The configuration file tells the FieldServer about its interfaces, and the routing of data required. In order to enable the FieldServer for McQuay Micro Tech Open Protocol communications, the driver independent FieldServer buffers need to be declared in the "Data Arrays" section, the destination device addresses need to be declared in the "Client Side Nodes" section, and the data required from the Servers needs to be mapped in the "Client Side Map Descriptors" section. Details on how to do this can be found below.

Note that in the tables, * indicates an optional parameter, with the **bold** legal value being the default.

4.1. Data Arrays

Section Title
Data_Arrays

Column Title	Function	Legal Values
Data_Array_Name	Provide name for Data Array	Up to 15 alphanumeric characters
Data_Format	Provide data format. Each Data Array can only take on one format.	Float, Bit, UInt16, SInt16, Packed_Bit, Byte, Packed_Byte, Swapped_Byte
Data_Array_Length	Number of Data Objects. Must be larger than the data storage area required for the data being placed in this array.	1-10,000

Example

```
//  Data Arrays
//
Data_Arrays
Data_Array_Name      ,Data_Format    ,Data_Array_Length
DA_AI_01             ,UInt16         ,200
DA_AO_01             ,UInt16         ,200
DA_DI_01             ,Bit            ,200
DA_DO_01             ,Bit            ,200
```

4.2. Client Side Connection Descriptors

Section Title
Connections

Column Title	Function	Legal Values
Port	Specify which port the device is connected to the FieldServer	P1-P8, R1-R2[1]
Baud*	Specify baud rate	300, 1200, 2400, 4800 and 9600 baud (Vendor limitation).
Parity*	Specify parity	Even (Vendor limitation)
Data_Bits*	Specify data bits	7 (Vendor limitation)
Stop_Bits*	Specify stop bits	**1**
Protocol	Specify protocol used	MCQ, McQuay
Handshaking*	Specify hardware handshaking	**None**
Poll Delay*	Time between internal polls	0-32000 seconds, **1 second**

Example

```
// Client Side Connections

Connections
Port        ,Baud       ,Parity     ,Data_Bits  ,Protocol   ,Handshaking  ,Poll_Delay
P1          ,9600       ,Even       ,7          ,McQuay     ,None         ,0.100s
```

[1] Not all ports shown are necessarily supported by the hardware. Consult the appropriate Instruction manual for details of the ports available on specific hardware.

4.3. Client Side Node Descriptors

Section Title
Nodes

Column Title	Function	Legal Values
Node_Name	Provide name for Node	Up to 32 alphanumeric characters
Node_ID	Network address of the device being polled. If the network address field is set to $00FF, the controller will respond to the packet without regard to its own internal network address	0-255
Route	This parameter must be used If units are connected to the FieldServer via a routing device e.g. OPM. The network address consists of the OPM address and the unit address.	
Protocol	Specify protocol used	MCQ, McQuay
Port	Specify which port the device is connected to the FieldServer	P1-P8, R1-R2[2]
PLC_Type	The name of the McQuay Equipment type being polled. Select one from the list on the left. The equipment type must be known for the driver to operate correctly.	200CFC, 100CFC, 050RPC, 001ASC, 000SCU, Type-Unknown
Password	Maximum of 8 characters.	Ascii characters only.

Example 1 of 2

```
//  Client Side Nodes
//  (No routing device used)

Nodes
Node_Name     ,Node_ID     ,Protocol     ,Port     ,Password     ,PLC_Type
Chiller       ,1           ,McQuay       ,P1       ,Abcdefgh     ,200CFC
```

Example 2 of 2

```
//  Client Side Nodes
//  (Routing device with address 3 used)

Nodes
Node_Name     ,Node_ID     ,Route     ,Protocol     ,Port     ,Password     ,PLC_Type
Chiller1      ,66          ,3         ,McQuay       ,P1       ,Abcdefgh     ,200CFC
```

[2] Not all ports shown are necessarily supported by the hardware. Consult the appropriate Instruction manual for details of the ports available on specific hardware.

4.4. Client Side Map Descriptors

4.4.1. FieldServer Related Map Descriptor Parameters

Column Title	Function	Legal Values
Map_Descriptor_Name	Name of this Map Descriptor	Up to 32 alphanumeric characters
Data_Array_Name	Name of Data Array where data is to be stored in the FieldServer	One of the Data Array names from “Data Array” section above
Data_Array_Offset	Starting location in Data Array	0 to maximum specified in “Data Array” section above
Function	Function of Client Map Descriptor	Rdbc, Wrbc, Wrbx

4.4.2. Driver Related Map Descriptor Parameters

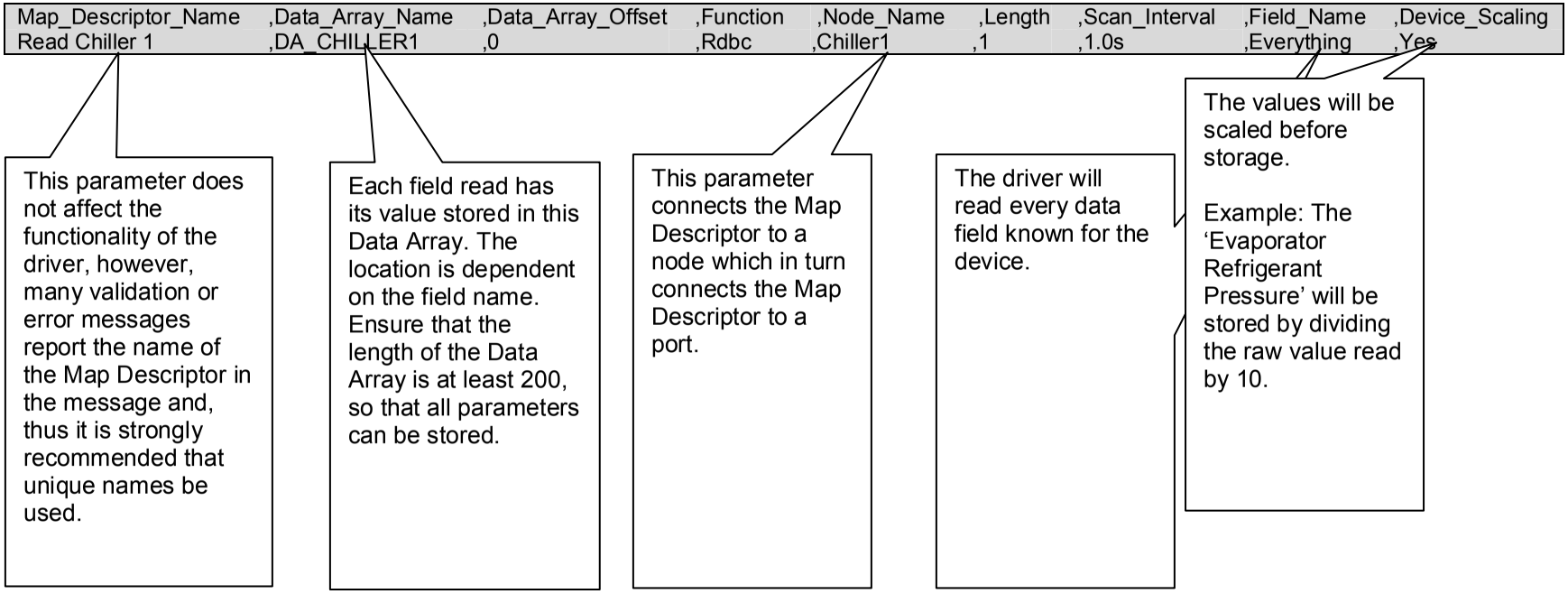
Column Title	Function	Legal Values
Node_Name	Name of Node to fetch data from	One of the node names specified in “Client Node Descriptor” above
Length	Length of Map Descriptor. Only one data field can be read or written per poll/response sequence, thus driver assumes a length of 1.	Set to 1 except when reading a direct address.
Field_Name	The name of the data field of interest. Each address with meaningful data in the McQuay device has a name. The field names are defined in a specification for each device type. If the field name is unknown, the Address can be defined. Refer to Appendix A.4 for further information.	‘Everything’ or select a field name from Table 2
Device_Scaling	If set to No then the driver stores the raw values, if set to Yes, the data is scaled before storing. The device scaling applied is shown in Table 1. Note that ‘Device_Scaling’ has no meaning when writing to a McQuay device.	Yes, **No**

4.4.3. Timing Parameters

Column Title	Function	Legal Values
Scan_Interval	Rate at which data is polled	>0.1s

4.4.4. Map Descriptor Example 1 – Read Everything.

In this example the Map Descriptor tells the driver to read all data fields known for the Node. The data fields are read sequentially and the value obtained placed in the Data Array at a location determined by the driver. Appendix A.2 outlines the data read and the Data Array location of each field. There is no way of telling this command to read some fields more often than others. Each field is read in turn and when the end of the list is reached the driver starts at the beginning again.



4.4.5. Map Descriptor Example 2 – Read a particular Data Field.

This example illustrates how to read data for one particular field of interest. The data is read every 2.0 seconds and the value obtained is stored in the Data Array named DA_OPERATIONAL_MODES at location 10.

Map_Descriptor_Name	,Data_Array_Name	,Data_Array_Offset	,Function	,Scan_Interval	,Node_Name	,Length	,Field_Name
Chiller10_Op_Mode	,DA_OPERATIONAL_MODES	,10	,Rdbc	,0s	,Chiller10	,1	,Chiller Operation Mode

Ensure that you spell and space the field name exactly as it is printed in table Appendix A.2

This Map Descriptor reads one single value from the McQuay device.

4.4.6. Map Descriptor Example 3 – Write.

This example illustrates how to write data to control fields in the McQuay device. This example illustrates the use of the Wrbx (write on change) function. A write message will be generated each time the data at index 0 in the Data Array changes. When writing multibyte fields, several messages are required to transfer the byte values to the McQuay devices. Note that 'Device_Scaling' has no meaning when writing to a McQuay device. The user must ensure that valid numbers are written to the data fields.

Map_Descriptor_Name	,Data_Array_Name	,Data_Array_Offset	,Function	,Node_Name	,Length	,Field_Name
Chiller1_Op_Mode	,DA_MODES	,0	,Wrbx	,Chiller1	,1	,Chiller Operation Mode

The data is only written when the contents of element zero of the array named DA_MODES changes.

The name of the data field to be written to.

5. Configuring the FieldServer as a McQuay Micro Tech Open Protocol Server

5.1. Server Side Connection Descriptors

Section Title
Connections

Column Title	Function	Legal Values
Port	Specify which port the device is connected to the FieldServer	P1-P8, R1-R2[3]
Baud*	Specify baud rate	300, 1200, 2400, 4800 and 9600 baud (Vendor limitation)
Parity*	Specify parity	Even (Vendor limitation)
Data_Bits*	Specify data bits	7 (Vendor limitation)
Stop_Bits*	Specify stop bits	**1**
Protocol	Specify protocol used	MCQ, McQuay
Handshaking*	Specify hardware handshaking	**None**

Example

```
// Server Side Connections

Connections
Port        ,Baud   ,Parity  ,Data_Bits  ,Protocol  ,Handshaking  ,Poll_Delay
P1          ,9600   ,Even    ,7          ,McQuay    ,None         ,0.100s
```

[3] Not all ports shown are necessarily supported by the hardware. Consult the appropriate Instruction manual for details of the ports available on specific hardware.

5.2. Server Side Node Descriptors

Section Title		
Nodes		
Column Title	**Function**	**Legal Values**
Node_Name	Provide name for node	Up to 32 alphanumeric characters
Node_ID	Network address of the device being polled. If the network address field is set to $00FF, the controller will respond to the packet without regard to its own internal network address	0-255
Route	This parameter must be used If units are connected to the FieldServer via a routing device e.g. OPM. The network address consists of the OPM address and the unit address.	
Protocol	Specify protocol used	MCQ, McQuay
Port*	Specify which port the device is connected to the FieldServer. This parameter is optional for Server side nodes. The reason is that when a poll is received on any port an attempt will be made to match it against any Map Descriptor/node irrespective of the port. If emulating more than one node with the same address, however, the Nodes must be linked to specific ports.	P1-P8, R1-R2[4]
PLC_Type	This parameter has no meaning for a Server but it must be specified to allow the driver to complete its configuration file validation.	200CFC, 100CFC, 050RPC, 001ASC, 000SCU, Type-Unknown
Password*	This parameter is ignored for a Server. Although every poll contains a password, this driver does no password validation when configured as a Server.	Ascii characters only.

Example

```
//  Server Side Nodes

Nodes
Node_Name       ,Node_ID    ,Protocol    ,PLC_Type
FieldServer     ,11         ,McQuay      ,200CFC
```

[4] Not all ports shown are necessarily supported by the hardware. Consult the appropriate Instruction manual for details of the ports available on specific hardware.

5.3. Server Side Map Descriptors

5.3.1. FieldServer Specific Map Descriptor Parameters

Column Title	Function	Legal Values
Map_Descriptor_Name	Name of this Map Descriptor	Up to 32 alphanumeric characters
Data_Array_Name	Name of Data Array where data is to be stored in the FieldServer	One of the Data Array names from “Data Array” section above
Data_Array_Offset	Starting location in Data Array	0 to maximum specified in “Data Array” section above
Function	Function of Client Map Descriptor	Passive

5.3.2. Driver Specific Map Descriptor Parameters

Column Title	Function	Legal Values
Node_Name	Name of Node to fetch data from	One of the node names specified in “Client Node Descriptor” above
Data_Type	This commonly used Field Server parameter has no meaning for this driver.	
Length	Length of Map Descriptor	1-10000
Address	Starting address of read block Addresses cannot be specified in hexadecimal format.	1 (Any positive integer)

5.3.3. Timing Parameters

Column Title	Function	Legal Values
Scada_Hold_Timeout	Specifies time Server side waits before responding to Client that node is offline on FieldServer Client side.	>1.0s

5.3.4. Map Descriptor Example – Strategy 1.

This example illustrates how you can configure the FieldServer as a McQuay Micro Tech Open Protocol Server. A single Map Descriptor is used to emulate a McQuay device's memory as one large single block of bytes. Any requests in the address range 1-0x270F can be serviced by this one Map Descriptor. The strategy is simple but has one drawback; a large Data Array which will be mostly empty is used. This does not affect performance but does use the FieldServer's memory resource.

When a poll is received, the driver looks through the Map Descriptor and finds one with a matching node and one whose address and length cover the poll's address.

Map_Descriptor_Name	,Data_Array_Name	,Data_Array_Offset	,Function	,Node_Name	,Address	,Length
Chiller1_emulation	,DA_CHILLER1	,0	,Passive	,Chiller1	,1	,10000

Ensure that Data Array's length is at least as long as the Map Descriptor's.

A data format of BYTE is appropriate,

Passive means that the Map Descriptor is used to respond to polls but does not generate any poll's itself.

The Node name connects the Map Descriptor to a Node definition and possibly to a port.

10,000 is the maximum length of a Data Array so there is no point in making the Map Descriptor longer.

5.3.5. Map Descriptor Example – Strategy 2.

This example differs from the previous example, in that it divides the memory being emulated into chunks to avoid having a large sparse array.

When a poll is received, the driver looks through the Map Descriptors and finds one with a matching node and one whose address and length cover the poll's address.

Map_Descriptor_Name	,Data_Array_Name	,Data_Array_Offset	,Function	,Node_Name	,Address	,Length
Chiller1_emulation1	,DA_CHILLER1	,0	,Passive	,Chiller1	,1024	,256
Chiller1_emulation2	,DA_CHILLER1	,256	,Passive	,Chiller1	,2048	,256

The same Data Array is used for both Map Descriptors.

By using the same Data Array and changing the offset the data can all be packed into a smaller array.

1024 = 0x400
2048 = 0x800

These two address ranges cover almost all the data fields in a 200 series chiller.

A few more Map Descriptors could be added to serve the remaining data field addresses.

Appendix A. Advanced Topics

Appendix A.1. Scaling

User Scaling:

The user can specify scaling in the configuration file which allows a value to be scaled before being stored in a Data Array or after being extracted from a data for writing to McQuay device. This scaling is specified by using the following keywords in Map Descriptor definition.

Data_Array_Low_Scale,
Data_Array_High_Scale,
Device_Low_Scale,
Device_High_Scale…..

Device Scaling:

The raw values read from a McQuay device may be treated as raw values or they may be scaled. Device scaling is the term used to describe the hard-coded scaling specified by McQuay for each data field. By applying device scaling the user can avoid details or knowledge of the scaling and used the converted value directly.

If this scaling is required then use the Device_Scaling parameter (see 4.4.2)

There are 5 device scaling methods. X refers to the raw value obtained from the McQuay device. The method that will be applied to each field is shown in the following table.

Table 1 – Device Scaling Method applied to each Field

Method Id.	Device Scaling Method
1	X
2	X / 10
3	(X – 1000) / 10
4	X / 2
5	X - 100
6	(X – 100) / 10
7	(X / 5) + 20
8	(X / 5) + 55
9	(X / 5)
10	(X / 5) - 100

Note that when device scaling is applied it is applied without consideration of the raw value. Where the raw value has a special meaning (available in the McQuay packet documents for the equipment type of interest); this driver does not apply any special consideration to these special values.

As a Client: - Device and User scaling may be applied.

As a Server. Device Scaling and User scaling is not applied. The raw value written to any data location is stored as is and the raw value read from a Data Array used in formatting the response to a read.

Appendix A.2. Data Field Name Tables[♣]

If you do not know the field name it is possible to read the device by specifying the address only. More information is provided in Appendix A.4

Table 2 – Equipment Type: 200CFC 200 Series Centrifugal Chiller

Array Location	Read/ Write/ Both	Field Name	Number of Bytes	Base Address	Device Scaling Method[5]
1	r	Model	3	0x0A08	1
4	r	Refrigerant	1	0x0A0B	1
5	r	Units of Measure	1	0x0A0C	1
6	r	Software Version	2	0x0A0D	1
8	r	Software Revision	1	0x0A0F	1
11	r	Chiller Unit Temperature Type	1	0x084B	1
12	r	Communication Status	1	0x040C	1
13	r	Compressor Lift Pressure	2	0x046A	2
15	r	Compressor Motor Current	2	0x044C	1
17	r	Compressor Motor Current Percent	1	0x044A	1
18	r	Compressor Number of Starts	2	0x085F	1
20	r	Compressor Operating Hours	2	0x0851	1
22	r	Compressor Suction Temperature	2	0x043A	3
24	r	Compressor Superheat - Discharge	2	0x046E	2
26	r	Compressor Superheat -Suction	2	0x046C	2
28	r	Condenser Approach Temperature	2	0x0484	2
30	r	Condenser Heat Recovery Unit Present	1	0x083F	1
31	r	Condenser Heat Recovery Temperature - Delta	2	0x048A	2
33	r	Condenser Heat Recovery Water Temp. - Entering	2	0x0454	3
35	r	Condenser Heat Recovery Water Temp. - Leaving	2	0x0456	3
37	r	Condenser Pump Status	1	0x0425	1
38	r	Condenser Pump # 1 Operating Hours	2	0x085A	1
40	r	Condenser Pump # 2 Operating Hours	2	0x085D	1
42	r	Condenser Refrigerant Pressure	2	0x0440	2
44	r	Condenser Refrigerant Temperature	2	0x0466	3
46	r	Condenser Subcooling Temperature	2	0x0480	2
48	r	Condenser Water Flow Rate	2	0x0452	1
50	r	Condenser Water Flow Status	1	0x045F	1
51	r	Condenser Water Rate Sensor	1	0x083D	1
52	r	Condenser Water Temperature - Delta	2	0x0488	2
54	r	Condenser Water Temperature - Entering	2	0x0436	3

[♣] Additional information may be obtained from McQuay International. Request the following document.
MicroTech® Series 200 Centrifugal Chiller Data Communications Packet for Open Protocol™ Cent2_22.doc

[5] See Appendix A.1.

Array Location	Read/ Write/ Both	Field Name	Number of Bytes	Base Address	Device Scaling Method[5]
56	r	Condenser Water Temperature - Leaving	2	0x0434	3
58	r	Cooling Tower Control	1	0x0926	1
59	r	Cooling Tower Stage	1	0x049A	1
60	r	Cooling Tower Valve Position	1	0x049B	1
61	r	Discharge Refrigerant Temperature	2	0x043E	3
63	r	Evaporator Approach Temperature	2	0x0482	2
65	r	Evaporator Pump Status	1	0x0423	1
66	r	Evaporator Pump #1 Operating Hours	2	0x0854	1
68	r	Evaporator Pump #2Operating Hours	2	0x0857	1
70	r	Evaporator Refrigerant Pressure	2	0x0438	2
72	r	Evaporator Refrigerant Temperature	2	0x0464	3
74	r	Evaporator Water Flow Rate	2	0x0450	1
76	r	Evaporator Water Flow Status	1	0x045E	1
77	r	Evaporator Water Rate Sensor	1	0x083C	1
78	r	Evaporator Water Temperature - Delta	2	0x0486	2
80	r	Evaporator Water Temperature - Entering	2	0x0432	3
82	r	Evaporator Water Temperature - Leaving	2	0x0430	3
84	r	Fault - Current Active	1	0x1C00	1
85	r	Last Start Hour	1	0x0861	1
86	r	Last Start Minute	1	0x0862	1
87	r	Last Start Month	1	0x0863	1
88	r	Last Start Date	1	0x0864	1
89	r	Last Start Year	1	0x0865	1
90	r	Last Stop Hour	1	0x0866	1
91	r	Last Stop Minute	1	0x0867	1
92	r	Last Stop Month	1	0x0868	1
93	r	Last Stop Date	1	0x0869	1
94	r	Last Stop Year	1	0x086A	1
95	r	Liquid Line Refrigerant Temperature	2	0x043C	3
97	r	Oil Pressure - Feed	2	0x0446	2
99	r	Oil Pressure - Net	2	0x0468	2
101	r	Oil Pressure - Vent	2	0x04AC	2
103	r	Oil Temperature - Feed	2	0x0442	3
105	r	Oil Temperature - Sump	2	0x0444	3
107	r	Outdoor Air Temperature - Network	1	0x0474	2
108	r	Refrigerant Detection Sensor	1	0x083E	1
109	r	Refrigerant Leak Detection Limit	1	0x044B	1
110	r	Unit Status	1	0x0420	1
111	b	Capacity Limit Percent	1	0x0475	1
112	b	Clear Current Fault	1	0x041E	1
113	b	Chiller Operation Mode	1	0x0477	1
114	b	Chilled Water Temperature Setpoint	1	0x0476	4

Array Location	Read/ Write/ Both	Field Name	Number of Bytes	Base Address	Device Scaling Method[5]
115	b	Communications Signal	1	0x047D	1
116	b	Outdoor Air Temperature - BAS	1	0x0473	5
117	b	Master/Server Setpoint	1	0x093C	1
118	b	Lead/Lag Mode Setpoint	1	0x093D	1
119	b	Enable Lag Setpoint	1	0x093E	1
120	b	Disable Lag Setpoint	1	0x093F	1
121	b	Lag Standby Setpoint	1	0x0942	1
122	b	Lead/Lag Switch Setpoint (Day)	1	0x0948	1
123	b	Lead/Lag Switch Setpoint (Hour)	1	0x0949	1
124	r	Lead/Lag Status	1	0x0401	1
125	r	Lead Unit	1	0x0402	1
126	r	Lead Unit Status	1	0x0403	1
127	r	Lag Unit Status	1	0x0404	1
128	r	Lead Motor Current	1	0x0405	1
129	r	Lag Motor Current	1	0x0406	1
130	r	Lead Status	1	0x0407	1
131	r	Lag Status	1	0x0408	1
132	b	Chiller Control mode	1	0x900	1
133	b	Chilled Water Temperature Local Setpoint	1	0x901	4
134	r	Chilled Water Temperature – Active Setpoint	1	0x905	4

Table 3 - Equipment Type: 050RPC Reciprocating Chiller

Array Location	Read/ Write/ Both	Field Name	Number of Bytes	Base Address	Device Scaling Method[6]
3	r	Circuit #1 Condenser Approach Temperature	2	0x0497	2
5	r	Circuit #1 Condenser Pressure	2	0x0467	2
7	r	Circuit #1 Alarm Conditions: Condenser Pressure	2	0x0822	2
9	r	Circuit #1 Alarm Conditions: Evaporator Pressure	2	0x0820	2
11	r	Circuit #1 Alarm Conditions: Liquid Line Temp	2	0x0826	6
13	r	Circuit #1 Alarm Conditions: Suction Temperature	2	0x0824	6
14	r	Circuit #1 Current Alarm	1	0x0831	1
16	r	Circuit #1 Evaporator Pressure	2	0x0463	2
18	r	Circuit #1 Liquid Line Temperature	2	0x0479	6
19	r	Circuit #1 Previous Alarm	1	0x0833	1
20	r	Circuit #1 Status	1	0x0428	1
22	r	Circuit #1 Subcooling Temperature	2	0x049B	2
24	r	Circuit #1 Suction Temperature	2	0x0475	6
26	r	Circuit #1 Superheat Temperature	2	0x0493	2
28	r	Circuit #2 Condenser Approach Temperature	2	0x0499	2

[6] See Appendix A.1.

Array Location	Read/ Write/ Both	Field Name	Number of Bytes	Base Address	Device Scaling Method[6]
30	r	Circuit #2 Condenser Pressure	2	0x0469	2
32	r	Circuit #2 Alarm Conditions: Condenser Pressure	2	0x082A	2
34	r	Circuit #2 Alarm Conditions: Evaporator Pressure	2	0x0828	2
36	r	Circuit #2 Alarm Conditions: Liquid Line Tempe	2	0x082E	6
38	r	Circuit #2 Alarm Conditions: Suction Temperature	2	0x082C	6
39	r	Circuit #2 Current Alarm	1	0x0832	1
41	r	Circuit #2 Evaporator Pressure	2	0x0465	2
43	r	Circuit #2 Liquid Line Temperature	2	0x047B	6
44	r	Circuit #2 Previous Alarm	1	0x0834	1
45	r	Circuit #2 Status	1	0x0429	1
47	r	Circuit #2 Subcooling Temperature	2	0x049D	2
49	r	Circuit #2 Suction Temperature	2	0x0477	6
51	r	Circuit #2 Superheat Temperature	2	0x0495	2
53	r	Compressor #1 Operating Hours	2	0x0862	1
55	r	Compressor #1 Starts	2	0x086A	1
57	r	Compressor #2 Operating Hours	2	0x0864	1
59	r	Compressor #2 Starts	2	0x086C	1
61	r	Compressor #3 Operating Hours	2	0x0866	1
63	r	Compressor #3 Starts	2	0x086E	1
65	r	Compressor #4 Operating Hours	2	0x0868	1
67	r	Compressor #4 Starts	2	0x0870	1
69	r	Condenser Entering Water Temperature	2	0x0470	6
71	r	Condenser Leaving Water Temperature	2	0x0472	6
73	r	Evaporator Entering Water Temperature	2	0x046E	6
75	r	Evaporator Leaving Water Temperature	2	0x0461	6
76	r	Heat/Cool Switch	1	0x08B4	1
78	r	OaT	2	0x047D	6
79	r	Stage of Capacity	1	0x042B	1
80	r	Unit % RLA	1	0x0474	1
81	r	Unit Status	1	0x0427	1
82	b	Clear Circuit #1 Alarm	1	0x091A	1
83	b	Clear Circuit #2 Alarm	1	0x091B	1
84	b	Condenser Leaving Water Temperature Setpoint	1	0x097D	1
85	b	Evaporator Entering Water Temperature Setpoint	1	0x090D	4
86	b	Evaporator Leaving Water Temperature Setpoint	1	0x0905	4
87	b	Maximum Water Reset Setpoint	1	0x090C	4
88	b	Network Command	1	0x044F	1
89	b	Network Demand Limit	1	0x044D	1
90	b	Network Leaving Water Temperature Reset	1	0x044E	1
91	b	Reset Option Setpoint	1	0x090B	1

Table 4 - Equipment Type: 100CFC 100 Series Centrifugal Chiller

Array Location	Read/ Write/ Both	Field Name	Number of Bytes	Base Address	Device Scaling Method[7]
1	r	Alarm Contact	1	0x0217	1
2	r	Approach Condenser	1	0x0418	1
3	r	Calculated Leaving Evaporator Temp Setpoint	1	0x0420	7
4	r	Calculated Network Amp Limit Setpoint	1	0x0423	1
5	r	Compressor Discharge Temp.	1	0x0407	1
6	r	Compressor Suction Temp. - Lo Resolution	1	0x0403	1
7	r	Compressor Suction Temp. - Hi Resolution	1	0x041C	7
8	r	Condenser Pump	1	0X021C	1
9	r	Condenser Refrigerant Pressure	1	0x0411	1
10	r	Condenser Refrigerant Temp.	1	0x0408	1
11	r	Condenser Water Flow Switch	1	0X050F	1
12	r	Cooling Tower Stage 1	1	0x0218	1
13	r	Cooling Tower Stage 2	1	0x0219	1
14	r	Current Fault	1	0x0428	1
15	r	Fault Last	1	0x0834	1
16	r	Fault 2	1	0x0835	1
17	r	Fault 3	1	0x0836	1
18	r	Fault 4	1	0x0837	1
19	r	Fault 5	1	0x0838	1
20	r	Fault 6	1	0x0839	1
21	r	Fault 7	1	0x083A	1
22	r	Fault 8	1	0x083B	1
23	r	Entering Condenser Water Temp. - Lo Resolution	1	0x0405	1
24	r	Entering Condenser Water Temp. - Hi Resolution	1	0x041E	8
25	r	Entering Evaporator Water Temp. - Lo Resolution	1	0x0401	1
26	r	Entering Evaporator Water Temp. - Hi Resolution	1	0x041A	1
27	r	Entering Water Reset	1	0x0438	1
28	r	Evaporator Low Pressure Switch	1	0x050E	1
29	r	Evaporator Pump	1	0X021B	1
30	r	Evaporator Refrigerant Pressure	1	0x0410	1
31	r	Evaporator Refrigerant Temp. - Lo Resolution	1	0x0402	1
32	r	Evaporator Refrigerant Temp. - Hi Resolution	1	0x041B	7
33	r	Evaporator Water Flow Switch	1	0X050F	1
34	r	High Discharge Pressure Switch	1	0X050E	1
35	r	Hot Gas Solenoid	1	0x0212	1
36	r	Leaving Condenser Water Temp. - Lo Resolution	1	0x0406	1
37	r	Leaving Condenser Water Temp. - Hi Resolution	1	0x041F	8
38	r	Leaving Evaporator Water Temp. - Lo Resolution	1	0x0400	1

[7] See Appendix A.1.

Array Location	Read/ Write/ Both	Field Name	Number of Bytes	Base Address	Device Scaling Method[7]
39	r	Leaving Evaporator Water Temp. - Hi Resolution	1	0x0419	7
40	r	LED Front Panel	1	0x021F	1
41	r	Lift Pressure	1	0x0412	1
42	r	Liquid Injection Solenoid	1	0x0216	1
43	r	Liquid Line Refrigerant Temp. - Lo Resolution	1	0x0404	1
44	r	Liquid Line Refrigerant Temp. - Hi Resolution	1	0x041D	8
45	r	Liquid Subcool	1	0x0417	1
46	r	Load Solenoid	1	0x0211	1
47	r	Low Evaporator Pressure Limit	1	0x043B	1
48	r	Manual Amp Limit	1	0X043D	1
49	r	Manual Amp Limit Setpoint	1	0x0901	1
50	r	Maximum Amp Limit	1	0X043C	1
51	r	Motor Control Latch Relay	1	0x0213	1
52	r	Motor Control Relay	1	0x021E	1
54	r	Motor Current Amps	2	0x0435	1
55	r	Motor Current Percent	1	0x040C	1
56	r	Motor Temperature Switch	1	0x050E	1
58	r	Number of Starts	2	0X0432	1
59	r	Oil Delta Pressure	1	0X040D	1
60	r	Oil Feed Temp.	1	0x0409	1
61	r	Oil Gage Pressure	1	0x0415	1
62	r	Oil Heater	1	0x0215	1
63	r	Oil Pressure Differential Switch	1	0X050E	1
64	r	Oil Pump	1	0X021D	1
65	r	Oil Sump Temp.	1	0X040A	1
67	r	Operating Hours	2	0x0430	1
68	r	Panel Rocker Switch	1	0X050E	1
69	r	Remote Amp Limit	1	0X043E	1
70	r	Remote Chilled Water Reset	1	0x0439	1
71	r	Remote Reset or Amp Limit Signal	1	0X040B	1
72	r	Remote Start/Stop Switch	1	0X050F	1
73	r	Soft Load Limit	1	0X043A	1
74	r	Starter Fault Switch	1	0X050F	1
75	r	Starter Transition Switch	1	0X050F	1
76	r	Superheat	1	0x0416	1
77	r	Surge Guard Switch	1	0X050E	1
78	r	Unit Status	1	0x0424	1
79	r	Unload Solenoid	1	0x0210	1
80	r	Vanes Closed Switch	1	0X050F	1
81	r	Watchdog Output	1	0x0214	1
82	b	Clear Current Fault	1	0X042A	1
83	b	Comm Port Baud Rate	1	0x0801	1
84	b	Correct Checksum	1	0x0800	1

Array Location	Read/ Write/ Both	Field Name	Number of Bytes	Base Address	Device Scaling Method[7]
85	b	Leaving Evaporator Temperature Setpoint	1	0x0900	7
86	b	Maximum Chilled Water Reset Setpoint	1	0x0820	9
87	b	Network Amp Reset	1	0x0450	1
88	b	Network Chilled Water Reset	1	0X044F	1
89	b	Network Start Stop	1	0x0451	1
90	b	Reset Option	1	0x0822	1
91	b	Soft Reset	1	0x0204	1
92	b	Start Mode	1	0x0824	1
93	b	Startup Differential Temp.	1	0x0826	9
94	b	Shutdown Differential Temp.	1	0x0827	9

Table 5 - Equipment Type : 001ASC Reciprocating Chiller

Array Location	Read/ Write/ Both	Field Name	Number of Bytes	Base Address	Device Scaling Method[8]
1	r	Active Chilled Water Temperature Setpoint	1	0x045A	4
2	r	Ckt#1 Condenser Pressure	2	0x0467	2
4	r	Ckt#1 Alarm Conditions: Capacity	1	0x1F13	1
5	r	Ckt#1 Alarm Conditions: Condenser Pressure	2	0x1F09	2
7	r	Ckt#1 Alarm Conditions: Evap Leaving Water Temp	2	0x1F0F	1
9	r	Ckt#1 Alarm Conditions: Evap Pressure	2	0x1F07	2
11	r	Ckt#1 Alarm Conditions: Fan Stage	1	0x1F14	1
12	r	Ckt#1 Alarm Conditions: Liquid Line Temperature	2	0x1F0D	10
14	r	Ckt#1 Alarm Conditions: Outside Air Temp	2	0x1F11	10
16	r	Ckt#1 Alarm Conditions: Suction Temperature	2	0x1F0B	10
18	r	Ckt#1 Current Alarm	1	0x08B9	1
19	r	Ckt#1 Current Alarm HourMinuteMonthDateYear	1	0x1F02	1
20	r	Ckt#1 Evaporator Pressure	2	0x0463	2
22	r	Ckt#1 Liquid Line Temperature	2	0x0479	10
24	r	Ckt#1 Status	1	0x0428	1
25	r	Ckt#1 Subcooling Temperature	2	0x04DF	2
27	r	Ckt#1 Suction Temperature	2	0x0475	10
29	r	Ckt#1 Superheat Temperature	2	0x04D7	2
31	r	Compressor #1 Operating Hours	2	0x0862	1
33	r	Compressor #1 Starts	2	0x086A	1
35	r	Evaporator Entering Water Temperature	2	0x046E	10
37	r	Evaporator Leaving Water Temperature	2	0x0461	10
39	r	Outdoor Air Temperature	2	0x047D	10
41	r	Stage of Cooling	1	0x042B	1
42	r	Unit Status	1	0x0427	1

[8] See Appendix A.1.

Array Location	Read/ Write/ Both	Field Name	Number of Bytes	Base Address	Device Scaling Method[8]
43	b	Clear Ckt#1 Alarm	1	0x091A	1
44	b	Evaporator Entering Water Temperature Setpoint	1	0x090D	2
45	b	Evaporator Leaving Water Temperature Setpoint	1	0x0905	2
46	b	Maximum Chilled Water Reset Setpoint	1	0x090C	2
47	b	Network Command	1	0x044F	1
48	b	Network Demand Limit	1	0x044D	1
49	b	Network Evaporator Leaving Water Temp Reset	1	0x044E	1
50	b	Reset Option Setpoint	1	0x090B	1

Table 6 - Equipment Type : Self-Contained Units (SCUs)

Array Location	Read/ Write/ Both	Field Name	Number of Bytes	Base Address	Device Scaling Method[9]
1	r	Air Velocity	1	0x0434	1
2	r	Airflow Status	1	0x045C	1
3	r	Current Alarm	1	0x0801	1
4	r	Previous Alarm	1	0x1301	1
5	r	Building Static Pressure	1	0x0435	1
6	r	Compressor #1 Alarm	1	0x081C	1
7	r	Compressor #1 Operating Hours	2	0x08AF	1
9	r	Compressor #2 Alarm	1	0x081D	1
10	r	Compressor #2 Operating Hours	2	0x08B2	1
12	r	Compressor #3 Alarm	1	0x081E	1
13	r	Compressor #3 Operating Hours	2	0x08B5	1
15	r	Compressor #4 Alarm	1	0x081F	1
16	r	Compressor #4 Operating Hours	2	0x08B8	1
18	r	Compressor #5 Alarm	1	0x0820	1
19	r	Compressor #5 Operating Hours	2	0x08BB	1
21	r	Compressor #6 Alarm	1	0x0821	1
22	r	Compressor #6 Operating Hours	2	0x08BE	1
24	r	Control Temperature	1	0x0432	1
25	r	Cool Stage	1	0x043F	1
26	r	Cooling Control Status	1	0x044A	1
27	r	Dew Point Temperature	1	0x042F	1
28	r	Duct Static Pressure #1	1	0x0435	1
29	r	Duct Static Pressure #2	1	0x0434	1
30	r	Economizer Enable	1	0x045A	1
31	r	Economizer Operating Hours	2	0x08C4	1
33	r	Economizer Position	1	0x042E	1
34	r	Entering Water Temperature	1	0x042C	1

[9] See Appendix A.1.

Array Location	Read/ Write/ Both	Field Name	Number of Bytes	Base Address	Device Scaling Method[9]
35	r	Fan - High Speed Operating Hours	2	0x08AC	1
37	r	Fan - Low Speed Operating Hours	2	0x08A9	1
39	r	Fan Operation	1	0x0462	1
40	r	Fan Speed	1	0x0436	1
41	r	Heat Stage	1	0x043E	1
42	r	Heating Control Status	1	0x044C	1
43	r	Heating Operating Hours	2	0x08C1	1
45	r	Leaving Water Temperature	1	0x042D	1
46	r	Mixed Air Temperature	1	0x042B	1
47	r	Motor Amps	1	0x0436	1
48	r	Motor Speed	1	0x04BB	1
49	r	Network Communications Status	1	0x0470	1
50	r	Outdoor Air Damper Status	1	0x045F	1
51	r	Outdoor Air Temperature	1	0x0429	1
52	r	Override Hours	2	0x08C7	1
54	r	Refrigerant Pressure	1	0x0431	1
55	r	Relative Humidity	1	0x0431	1
56	r	Return Air Temperature	1	0x0428	1
57	r	Space Temperature	1	0x042A	1
58	r	Supply Air Temperature	1	0x0427	1
59	r	Supply Fan Status	1	0x045E	1
60	r	Unit Enabled Status	1	0x0488	1
61	r	Unit Status	1	0x043D	1
62	r	Variable Inlet Vane Position	1	0x0436	1
63	r	Water Flow Required	1	0x0445	1
64	r	Water Flow	1	0x045B	1
65	r	Water Pump Start/Stop	1	0x0461	1
66	b	Building Static Pressure Setpoint	1	0x0923	1
67	b	Clear Alarm	1	0x04CE	1
68	b	Clear Alarm Buffer	1	0x048A	1
69	b	Control Mode	1	0x044F	1
70	b	Control Temperature Source	1	0x0926	1
71	b	Cooling Control Deadband	1	0x088D	1
72	b	Cooling Control Setpoint	1	0x0907	1
73	b	Cooling Supply Deadband	1	0x0878	1
74	b	Cooling Supply Setpoint	1	0x090E	1
75	b	Dew Point Deadband	1	0x094E	1
76	b	Dew Point Sensor Type	1	0x0949	1
77	b	Dew Point Setpoint	1	0x094D	1
78	b	Duct Static Pressure Setpoint	1	0x0917	1
79	b	Duct Static/Building Static Pressure Deadband	1	0x089B	1
80	b	Economizer (Air) Changeover Differential	1	0x0932	1
81	b	Economizer (Air) Changeover Temperature Setpoint	1	0x0931	1

Array Location	Read/ Write/ Both	Field Name	Number of Bytes	Base Address	Device Scaling Method[9]
82	b	Economizer (Air) Method	1	0x083B	1
83	b	Economizer (Water) Changeover Temperature Offset	1	0x091A	1
84	b	Fan On Heating Setpoint	1	0x0939	1
85	b	Head Pressure Deadband	1	0x08E1	1
86	b	Head Pressure Setpoint	1	0x093A	1
87	b	Heating Control Deadband	1	0x0894	1
88	b	Heating Control Setpoint	1	0x0905	1
89	b	Heating Supply Deadband	1	0x0871	1
90	b	Heating Supply Setpoint	1	0x0914	1
91	b	Humidity Control Type	1	0x0947	1
92	b	Humidity Deadband	1	0x094C	1
93	b	Humidity Setpoint	1	0x094B	1
94	b	Low Entering H20 Comp. Lockout Differential	1	0x092C	1
95	b	Low Entering Water Compressor Lockout Setpoint	1	0x092B	1
96	b	Min Vane Position/Fan Speed	1	0x0951	1
97	b	Minimum Outdoor Air Damper Position	1	0x0934	1
98	b	Minimum Supply Air Temperature Control	1	0x0919	1
99	b	Unoccupied Cooling Space Differential	1	0x0904	1
100	b	Unoccupied Cooling Space Setpoint	1	0x0903	1
101	b	Unoccupied Heating Space Differential	1	0x0902	1
102	b	Unoccupied Heating Space Setpoint	1	0x0901	1
103	b	Water Flow Signal	1	0x0444	1

Appendix A.3. McQuay Equipment Types recognized by the driver.

Code	Description
200CFC	200 Series Centrifugal Chiller
100CFC	100 Series Centrifugal Chiller
001ASC	Air Screw Chiller
000SCU	Self Contained Unit
0050RPC	Reciprocating Chiller

Appendix A.4. Direct Addressing

Direct addressing allows data to be read from the McQuay device when the field name is unknown. McQuay specifies the addresses in hexadecimal notation but they must be specified in the CSV file in decimal format. For example, to read McQuay address 0x0400, the address in the CSV file must be entered as 1024.

If reading multiple addressing at once (i.e. if the length is greater than 1) it may be necessary to adjust the timeout for the Map Descriptor.

It is possible to read multiple addresses using one Map Descriptor; however, it is only possible to write to a single address.

The following Map Descriptor parameters will need to be defined in addition to those described in section 4.4.

5.3.6. Driver Related Map Descriptor Parameters

Address*	The address in the McQuay device that contains the data of interest.	Normally this field is set to 1 or omitted.
Bytes_Per_Field*	This parameter is used with direct addressing. When reading, up to 4 consecutive address locations in the McQuay device can be combined and stored in a single Data Array element. When writing, one value from the Data Array may be sent to up to 4 consecutive address locations in the McQuay device.	1,2,3,4

Refer to examples on the following pages.

Appendix A.5. Advanced Map Descriptor Example 1– Reading direct address.

This example reads 10 data elements from a McQuay device starting at address 4096 (= 0x1000). The 10 values obtained are stored in DA_CHILLER1 starting at offset zero.

Map_Descriptor_Name	,Data_Array_Name	,Data_Array_Offset	,Node_Name	,Address	,Length	,Scan_Interval	,Function
Chiller1_emulation	,DA_CHILLER1	,0	,Chiller1	,4096	,10	,1.0s	,Rdbc

A data format of BYTE is appropriate,

Specify the address in decimal

Appendix A.6. Advanced Map Descriptor Example 2 – Controlling Bytes per Field – Reading

This example reads a McQuay device starting at address 4096 (= 0x1000) to 4099 incl. Before storing the data from the responses, the FieldServer combines the 4 values into a single value and stores this single value in a single Data Array element.

The driver reads address 4096 and gets a value. Call this value v0.
The driver reads address 4097 and gets a value. Call this value v1.
The driver reads address 4098 and gets a value. Call this value v2.
The driver reads address 4099 and gets a value. Call this value v3.
When the 4th address has bee read, the driver calculates Total_Value = v0 + v1 * 0x100 + v2 * 0x10000 + v3 * 0x1000000. The Total_Value is stored in the Data Array.

Map_Descriptor_Name	,Data_Array_Name	,Data_Array_Offset	,Node_Name	,Address	,Length	,Bytes_per_Field	,Scan_Interval	,Function
Chiller1_emulation	,DA_CHILLER1	,0	,Chiller1	,4096	,1	,4	,1.0s	,Rdbc

Specify the address in decimal

Length must be set to 1 when Bytes_per_Field is specified.

A value between 1 and 4 - tells the driver how many consecutive addresses to read from the remote device. The data value for each address is combined into a single value before it is stored.

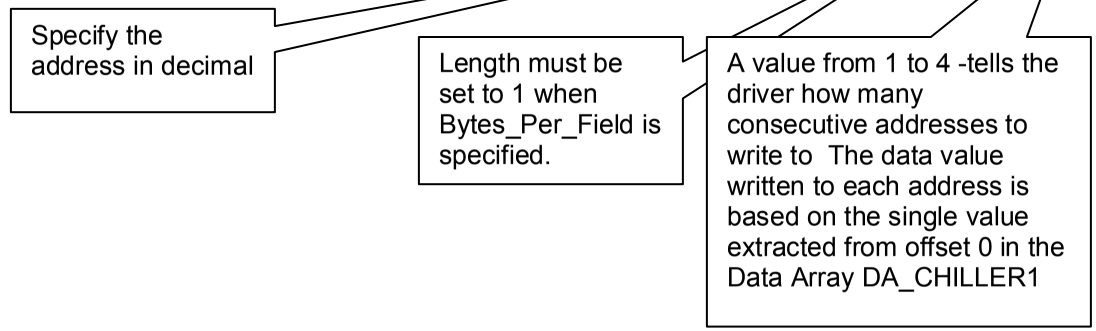
Appendix A.7. Advanced Map Descriptor Example 3 – Controlling Bytes per Field – Writing

This example writes to a McQuay device starting at address 4096 (= 0x1000) to 4097incl. The value written to each address location is based on a single value extracted from the FieldServer's Data Arrays.

The value v0 = (value_extracted_from_DA) AND 0xff
The value v1 = (value_extracted_from_DA shift right by 8) AND 0xff

The driver writes value v0 to address 4096.
The driver writes value v1 to address 4097.

Map_Descriptor_Name	,Data_Array_Name	,Data_Array_Offset	,Node_Name	,Address	,Length	,Bytes_Per_Field	,Scan_Interval	,Function
Chiller1_emulation	,DA_CHILLER1	,0	,Chiller1	,4096	,1	,2	,1.0s	,Wrbc



Appendix A.8. Statistics

The driver reports statistics according to the FieldServer standards. The following notes describe some aspects of standard statistic reporting which are peculiar to this driver.

NAK	Each time a negative acknowledgement message is received.
CHECKSUM	Each time that a message is received that contains a non-ASCII character.
NOISE	An acknowledgement message contains an unrecognized code.
NO START	Each time a message is received that doesn't begin with a space.
PROTOCOL	All other errors are reported as protocol errors

In addition to the standard FieldServer communication statistics described above and in the FieldServer User's Manual, this driver can also expose some driver statistics by writing data to a Data Array. A special Map Descriptor is required. The driver recognizes the Map Descriptor by its name which must be "McQuay-stats".

The driver stores the following data. The location in the Data Array is obtained by multiplying the port number by 50 and then using the location offset indicated in the table below.

Location	Statistic
0	MCQ_STAT_PASSWORD
10	MCQ_STAT_NAK
11	MCQ_STAT_NOISE
12	MCQ_STAT_NOISE_CODE
13	MCQ_STAT_NON_ASCII_CHARS
14	MCQ_STAT_BAD_START_BYTE
15	MCQ_STAT_OTHER_CMPLT_ERR
16	MCQ_STAT_CLIENT_BYTES_RCVD
17	MCQ_STAT_CLIENT_FRAGS_RCVD
18	MCQ_STAT_CLIENT_MSGSS_RCVD
19	MCQ_STAT_CLIENT_BYTES_SENT
20	MCQ_STAT_CLIENT_FRAGS_SENT
21	MCQ_STAT_CLIENT_MSGSS_SENT
22	MCQ_STAT_SERVER_BYTES_SENT
23	MCQ_STAT_SERVER_MSGSS_SENT
24	MCQ_STAT_SERVER_NAKS_SENT

The following example shows how this special Map Descriptor can be configured. You can copy this section of text directly into your CSV file.

```
Nodes
Node_Name          ,Station          ,Port             ,Protocol   ,PLC_Type
Mcq_stats          ,0                ,P1               ,McQuay     ,200CFC

Data_Arrays
Data_Array_Name    ,Data_Format      ,Data_Array_Length
DA_MCQ_STATS       ,UINT32           ,2000

Map_Descriptors
Map_Descriptor_Name ,Data_Array_Name ,Data_Array_Offset ,Function ,Node_Name ,Address ,Length
McQuay-Stats       ,DA_MCQ_STATS     ,0                ,Passive  ,Mcq_stats ,1       ,2000
```

When the driver sees this Map Descriptor it uses the Data Array DA_MCQ_STATS (in this example) to store driver specific statistics. Only one of these Map Descriptors may be specified per FieldServer.

Appendix B. Error Messages

Those messages marked with an * are only printed once even if they occur repeatedly.

Error Message	Action
McQuay:#1 FYI. The MapDesc called <%s> is too short	The length of the Map Descriptor used to expose driver statistics is too short. Set the length to at least 1000. You can ignore this message – the driver will abandon excess statistics.
McQuay:#2 FYI. You could have used a MapDesc called <%s> to expose diagnostic info.	You can safely ignore this message. It is a prompt. Refer to Appendix A.8.
McQuay:#3 FYI. Normally passwords are defined for each node.	This is a reminder that a password is expected when a node is defined. Refer to sections 4.3 or 5.2. Even though password checking is not enforced when the driver is configured as a Server, the driver expects one to be defined for every node.
McQuay:#4 Err. Node list if full. Max=%d	A maximum of 100 McQuay device nodes can be configured per FieldServer. If the limit is reached, call support.
McQuay:#5 Err. Node=%d has more than one password.	A node has been provided with two passwords. This is either a configuration error or two nodes with the same address are connected to different ports. The driver can only store one password per node address and cannot use the port number to differentiate them. Re-program the McQuay devices to have the same passwords if they have the same addresses.
*McQuay:#9 FYI. Device scaling method unknown.	Device Scaling is being applied and the driver doesn't know how to scale a variable. Please submit your configuration file with your request for support.
McQuay:#10 Err. Node=%s. Equip. type not recognized.	The driver doesn't recognize the equipment type specified using the PLC_Type parameter in the configuration file. Refer to Appendix A.3 for a list of valid equipment types.[10]
McQuay:#11 Err. Node=%s. Equip. Type not specified.	Refer to Sections 4.3, 5.2. and Appendix A.3[10]
McQuay:#13 Err. MapDesc=%s. Field Name unknown.	Refer to Appendix A.2 for a list of valid field names. Refer also to Sections 4.4 or 5.3.
McQuay:#14 Err. MapDesc=%s. Max Len=1 for Writes with direct addr.	Corrective action is required. When writing to a McQuay device using direct addressing the length may only be 1[10]
McQuay:#15 Err. MapDesc=%s. Field Name/Address Required.	Neither a field name nor an address was specified. The driver does not know what location to read. Specify a field name from Appendix A.2.[10]
McQuay:#16 Err. MapDesc=%s. No Node.	Each Map Descriptor must be connected to a node. This is done by specifying a node name. [10]
McQuay:#17 Err. MapDesc=%s. Md too short.	The offset added to the length of the Map Descriptor extends beyond the Data Array. Increase the Data Array length. [10]
McQuay:#18 Err. MapDesc=%s. Cannot write 'Everything'.	When using 'everything' as a field name the Map Descriptor function must be Rdbc or Rdb as this is a read only function.[10]
McQuay:#19 do diagnostic 3	Call FieldServer Technical Support. A developer diagnostic has been called and should not have been.
McQuay:#20 do diagnostic 1	
McQuay:#21 do diagnostic 2	

[10] Correct the configuration by editing the CSV file and downloading it to the FieldServer, then reset the FieldServer for the changes to take effect.

Error Message	Action
McQuay:#22 Err. Illegal Node_ID [%d] - Set to 1	Check configuration file, a Node_ID is out of range. [11]
McQuay:#23 FYI. Config requires non-critical update. Read Manual.	If the Node_ID is greater than 255 then this message is printed. In versions of the driver prior to 1.02 this was permitted and these configurations will operate correctly. In versions since 1.02a the route parameter is used to provide the routing device's address. Convert the Node_ID as follows. Divide the old Node_ID by 256. Set the Node_ID equal to remainder and set the route equal to the quotient. Example; Old Node_ID = 834. 834 div 256 = 3 remainder 66. Therefore Route=3 and Node_ID=66.
McQuay:#24 Err. Illegal Node_ID/Route	If the Node_ID is > 255 and the Route is specified too then this error is printed. The Node_ID should be a number in the range 0 to 255.**Error! Bookmark not defined.**
McQuay:#25 Err. Max Len=1 when Bytes_Per_Field > 1. MD=%s.	The length parameter must be set to 1 when a Map Descriptor has the Bytes_Per_Field parameter specified[10]
McQuay:#26 Err. Max Bytes_Per_Field=4. MD=%s.	Legal values for the Bytes_Per_Field parameter are whole numbers in the range 1 to 4 inclusive. [10]
McQuay:#27* Err. Response=NAK. Read Manual. Maybe password or Node_ID	This message is printed if a NAK response is received in the first few polls to a node. The driver guesses that the reason is that the Node_ID/route or password has been incorrectly specified in the configuration file. Verify these settings. The message is printed once and suppressed for subsequent occurrences. Refer to Appendix C.2
McQuay:#28* Err. Device responded with a NAK.	The message is printed and then suppressed for subsequent occurrences, but the NAK stat is incremented for each occurrence. The driver is reporting that the McQuay device responded with a NAK under different circumstances from msg#27 which is printed if the NAK is received during the first few polls. The message indicates that the McQuay device could not respond. If the NAK's are occasional, assume that noise has corrupted an occasional message. If they occur frequently assume a systematic or connection error. Refer to **Error! Reference source not found.** and Appendix C.2.

[11] Correct the configuration by editing the CSV file and downloading it to the FieldServer, then reset the FieldServer for the changes to take effect.

Appendix C. Troubleshooting

Appendix C.1. Connection Problems

The driver produces a timeout each time a message is sent. If the number of timeouts is the same as the number of messages sent then you know that the McQuay device has never sent a response. The following reasons may be applicable:

- Incorrect connection settings - the messages sent by the FieldServer cannot be interpreted by the McQuay device.
- Incorrect connection wiring – the McQuay device may require a jumper on its serial port to deflect hardware handshaking. Refer to the vendor manual.
- Serial port failure – If the Tx LED is not flashing each time a message is sent, the port is not working
- The McQuay device is off.
- If the Node_ID and Route are incorrect, the McQuay device will not respond at all.

Appendix C.2. Negative Acknowledgement - NAK

The Server sends a NAK message and increments the NAK stat each time a poll is unsuccessful. An occasional NAK may indicate a corrupted message. If the number of NAK's is the same as the number of transmitted messages to a Node, however, one of the following problems could apply:

- Bad Password - the password specified in the "read" command did not match any of the access level passwords stored in the MicroTech controller to which the terminal is connected. Call FST Tech support for default passwords or call your Vendor. FST recommends checking the password first as in almost all reports the problem was resolved by changing the password.
- Device has been polled with an invalid command code.
- Bad Node_ID / Route parameters for the node.
- The MCQ device received the message and thought it was badly formatted. This is unlikely on a repeated basis unless the connection settings were slightly wrong.
- The packet structure was invalid
- The packet was received with a parity or framing error

Appendix C.3. Node ID problems

If the Node is incorrectly specified, the FieldServer will not get a response from the McQuay unit. The ID used to communicate with the McQuay unit needs to be determined. This may not be the same as shown on the Rotary switches of the unit. Note that McQuay uses Hexadecimal notation for addressing, therefore, if connection to the Unit is with address 0201, this is in Hex and the Node ID for the FieldServer configuration needs to be 513.

Appendix C.4. Server Side Configuration – Consecutive Addresses

Some variables have their values stored in two consecutive address locations. When the Client polls for the value it actually sends two read messages – one for each address. When configuring the Server side, therefore, it is necessary to define two Map Descriptors for variables which require two consecutive addresses.

Map_Descriptors						
Map_Descriptor_Name	,Data_Array_Name	,Data_Array_Offset	,Function	,Node_Name	,Address	,Length
Supply Fan Status	,DA_AI_01	,01	,Passive	,Node_A	,1118	,1
Supply Temp	,DA_AI_01	,02	,Passive	,Node_A	,1063	,1
Space Temp	,DA_AI_01	,03	,Passive	,Node_A	,1066	,1
Control Temp Part 1	,DA_AI_01	,04	,Passive	,Node_A	,1074	,1